In the Claims

1. (Currently Amended) A computer implemented replacement selection method for <u>merging</u> ~~organizing~~ data items from two or more input streams comprising:

assigning a status identifier to each input stream, said identifier reflecting a state of an input stream;

comparing a status identifier of a first input stream with a status identifier of a second input stream;

identifying a data item being processed from a second input stream as being a duplicate of a previously processed data item from a first input stream based on said assigned status identifier;

assigning a duplicate status identifier to said second input stream responsive to said identified duplicate data item; and

~~switching~~ <u>processing a value</u> from ~~said first input stream to~~ said second input stream responsive to said assigned duplicate status identifier, wherein the step of ~~switching~~ <u>processing said value</u> ~~said input streams~~ <u>merges said input streams into a single output stream and</u> avoids exhausting one of said input streams when a quantity of said input streams is an odd number <u>greater than one</u>.

2. (Previously Presented) The method of claim 1, wherein the indication that the data item being processed is a duplicate data item is one value of an indicator having values the status identifier has a value corresponding to "empty", "duplicate", "merging" and "done".

3. (Previously Presented)   The method of claim 2, wherein:

the status identifier is an integer variable;

the status identifier value corresponding to "empty" is the value zero;

the status identifier value corresponding to "duplicate" is the value one;

the status identifier value corresponding to "merging" is the value two; and

the status identifier value corresponding to "done" is the value three.

2

4. (Currently Amended) The method of claim 3, wherein the step of ~~alternating between~~ processing a value from said input ~~streams~~ stream containing said duplicate item is responsive to comparisons between the values of the integer status identifier values associated with data items being compared.

5. (Original)   The method of claim 1, wherein the method is a replacement selection method using a loser- oriented selection tree.

6. (Currently Amended)   A computer-readable data structure representing a selection tree for use in a computer- implemented replacement selection method of merging ~~organizing~~ data items from two or more input streams, comprising
      for each node of the selection tree:
            an identifier of one of the input streams, and
            a reference to a data item being processed from that one of the input
      streams;
            instructions to compare identifiers from a node of a first input stream with a node of a second input stream, and in response to said identifiers of said second input stream representing a duplicate data item assigning said identifier of said second input stream as a duplicate identifier and ~~switching~~ processing a value from ~~said first input stream to~~ said second input to avoid exhausting said first input stream when a quantity of said input streams is an odd number greater than one.

7. (Original)   The data structure of claim 6, wherein the indication that the data item being processed is a duplicate is one value of an indicator having values corresponding to "empty", "duplicate", "merging" and "done".

8. (Original)   The data structure of claim 7, wherein:
      the indicator is an integer variable;
      the indicator value corresponding to "empty" is the value zero,
      the indicator value corresponding to "duplicate" is the value one;

3

the indicator value corresponding to "merging" is the value two; and

the indicator value corresponding to "done" is the value three.

9. (Currently Amended)   An article for use in a computer implemented replacement
selection method for ~~organizing~~ merging data items from two or more input streams
comprising:

a computer-readable signal-bearing medium;

means in the medium for assigning a status identifier to each input stream, said
identifier reflecting a state of an input stream;

means in the medium for comparing a status identifier of a first input stream with
a status identifier of a second input stream;

means in the medium for identifying a data item being processed from one of the
input streams as being a duplicate of a previously processed data item;

means in the medium for assigning a duplicate status identifier to said second
input stream responsive to said identified duplicate data item; and

means in the medium for processing a value from ~~switching from said first input
stream to~~ said second input stream responsive to said assigned duplicate status identifier,
wherein the step of processing said value ~~switching said input streams~~ merges said input
streams into a single output stream and avoids exhausting one of said input streams when
a quantity of said input streams is an odd number greater than one.

10. (Original) The article of claim 9, wherein the indication that the data item being processed is
a duplicate data item is a value of an indicator having values corresponding to "empty",
"duplicate", "merging" and "done".

11. (Original) The article of claim 10, wherein the indicator is an integer variable;
the indicator value corresponding to "empty" is the value zero;
the indicator value corresponding to "duplicate" is the value one;
the indicator value corresponding to "merging" is the value two; and

4

the indicator value corresponding to "done" is the value three.

12. (Currently Amended) The article of claim 11, wherein the means for ~~switching~~ <u>processing</u> is responsive to comparisons between the values of the integer variable indicator values associated with data items being compared.

13. (Original) The article of claim 9, wherein the method is a replacement selection method using a loser- oriented selection tree.

14. (Original) The article of claim 9, wherein the medium is selected from the group consisting of a recordable data storage medium; and a modulated carrier signal.

5